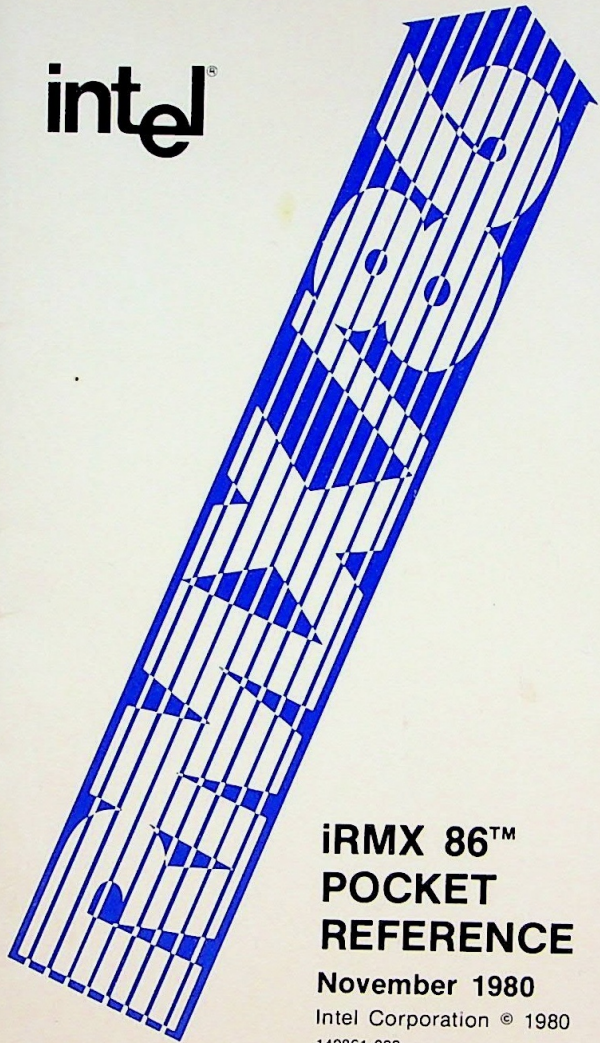


intel[®]



**iRMX 86[™]
POCKET
REFERENCE**

November 1980

Intel Corporation © 1980

142861-002

SUMMARY OF SYSTEM CALLS

Nucleus System Calls As They Apply To Object Types:

Jobs	Tasks	Segments	Mailboxes	Semaphores	Regions	Extensions	Composites
Catalog\$object Uncatalog\$object							
Creates\$- job Deletes\$- job Offspring	Creates\$- task Deletes\$- task Suspend\$task	Creates\$- segment Deletes\$- segment Get\$size	Creates\$- mailbox Deletes\$- mailbox Send\$- message	Creates\$- semaphore Deletes\$- semaphore Receive\$- units	Creates\$- region Deletes\$- region Receive\$- control	Creates\$- extension Deletes\$- extension	Creates\$- composite Deletes\$- composite Inspect\$- composite Alter\$- composite
Get\$pool\$- attributes Set\$pool\$- minimum	Resume\$task Sleep Get\$task\$- tokens Set\$priority Get\$priority		Receive\$- message	Send\$- units	Send\$- control Accept\$- control		
Jobs	Tasks	Segments	Mailboxes	Semaphores	Regions	Extensions	Composites

Asynchronous I/O System Calls As They Apply To File Types:

File Type

System Call	Stream	Physical	Named Data	Named Directory
Attach\$file
Change\$access
Close
Create\$directory
Create\$file
Delete\$connection
Delete\$file
Get\$connection\$status
Get\$directory\$entry
Get\$extension\$data
Get\$file\$status
Get\$path\$component
Open
Read
Rename\$file
Seek
Set\$extension\$data
Special
Truncate
Write

SYSTEM CALLS

The following abbreviations are used to indicate the data types of parameters in the system calls shown later:

Abbreviation

Meaning

B

Byte

W

Word not containing a token

WT

Word containing a token

DW

Double word

PW

Pointer to a word not containing a token

PWT

Pointer to a word containing a token

PS

Pointer to a string

PD

Pointer to a data structure

PI

Pointer to an instruction

P

Any other kind of pointer

A blue asterisk (*) following the semicolon indicates that a system call is for system programmers only

job\$flags, task\$priority, start\$address, data\$seg, stack\$ptr,
W B PI WT P

stack\$size, task\$flags, except\$ptr);
W W PW

Excpt\$hdlr\$info:

STRUCTURE(EXCEPTION\$HANDLER\$OFFSET WORD,
EXCEPTION\$HANDLER\$BASE WORD,
EXCEPTION\$MODE BYTE);

Exception\$Mode	Control to Exception Handler
0	Never
1	On programmer error only
2	On environmental condition only
3	On all exceptional conditions

Job\$flags	Meaning
0	No parameter validation for new job
2	Parameter validation for new job

ESOK, E\$CONTEXT, E\$EXIST, E\$LIMIT, E\$MEM, E\$PARAM

mailbox = RQ\$CREATE\$MAILBOX (mailbox\$flags, except\$ptr);
WT W PW

mailbox\$flags	Queueing Scheme
0	First-in-first-out
1	Priority Based

ESOK, E\$LIMIT, E\$MEM, E\$NOT\$CONFIGURED

region = RQ\$CREATE\$REGION (region\$flags, except\$ptr);
WT W PW

Region\$flags	Queueing Scheme
0	First-in-first-out
1	Priority based

ESOK, E\$LIMIT, E\$MEM, E\$NOT\$CONFIGURED

segment = RQ\$CREATE\$SEGMENT (size, except\$ptr);
WT W PW

ESOK, E\$LIMIT, E\$MEM, E\$NOT\$CONFIGURED

semaphore = RQ\$CREATE\$SEMAPHORE (initial\$value,
WT W

max\$value, sem\$flags, except\$ptr);
W W PW

sem\$flags

Queueing Scheme

0

First-in-first-out

1

Priority Based

ESOK, ESLIMIT, E\$MEM, E\$PARAM, E\$NOT\$CONFIGURED

task = RQ\$CREATE\$TASK (priority, start\$address, data\$seg,
WT B PI W

stack\$ptr, stack\$size, task\$flags, except\$ptr);
P W W PW

ESOK, ESLIMIT, E\$MEM, E\$NOT\$CONFIGURED, E\$PARAM

CALL RQ\$DELETE\$COMPOSITE (extension, composite,
WT WT

except\$ptr); *
PW

ESOK, E\$CONTEXT, E\$EXIST, E\$MEM, E\$NOT\$CONFIGURED,
E\$TYPE

CALL RQ\$DELETE\$EXTENSION (extension, except\$ptr); *
WT PW

ESOK, E\$EXIST, E\$MEM, E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$DELETE\$JOB (job, except\$ptr);
WT PW

ESOK, E\$CONTEXT, E\$EXIST, E\$MEM, E\$NOT\$CONFIGURED,
E\$TYPE

CALL RQ\$DELETE\$MAILBOX (mailbox, except\$ptr);
WT PW

ESOK, E\$EXIST, E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$DELETE\$REGION (region, except\$ptr); *
WT PW

ESOK, E\$CONTEXT, E\$EXIST, E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$DELETE\$SEGMENT (segment, except\$ptr);
WT PW

E\$OK, E\$EXIST, E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$DELETE\$SEMAPHORE (semaphore, except\$ptr);
WT PW

E\$OK, E\$EXIST, E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$DELETETASK (task, except\$ptr);
WT PW

E\$OK, E\$CONTEXT, E\$EXIST, E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$DISABLE (level, except\$ptr);
W PW

level:	Bits	Value
	15-7	0
	6-4	the interrupt level (0-7)
	3	1
	2-0	0

Note that actual level n equals encoded level n8H.

E\$OK, E\$CONTEXT, E\$NOT\$CONFIGURED, E\$PARAM

CALL RQ\$DISABLE\$DELETION (object, except\$ptr);
WT PW

E\$OK, E\$EXIST, E\$LIMIT, E\$NOT\$CONFIGURED

CALL RQ\$ENABLE (level, except\$ptr);
W PW

level:	Bits	Value
	15-7	0
	6-4	the interrupt level (0-7)
	3	1
	2-0	0

Note that actual level n equals encoded level n8H.

E\$OK, E\$CONTEXT, E\$NOT\$CONFIGURED, E\$PARAM

ESOK, E\$CONTEXT, E\$EXIST, E\$NOT\$CONFIGURED

Note that actual level n equals encoded level $n8H$.

ESOK, ECONTEXT, E\$NOT\$CONFIGURED, E\$PARAM

Note that actual level n equals encoded level $n8H$.

ESOK. ESPARAM

ESOK, EEXIST, ESMEM, E\$NOT\$CONFIGURED, E\$TYPE

Exception\$Info:

```
STRUCTURE(EXCEPTION$HANDLER$OFFSET WORD,
          EXCEPTION$HANDLER$BASE WORD,
          EXCEPTION$MODE BYTE);
```

Exception\$Mode	Control to Exception Handler
0	Never
1	On programmer error only
2	On environmental condition only
3	On all exceptional conditions

E\$OK, E\$NOT\$CONFIGURED

level = RQ\$GET\$LEVEL (except\$ptr);

W

PW

level:	Bits	Value/Interpretation
	15-8	undefined
	7	0 some level is being serviced and bits 6-4 are significant
		1 no level is being serviced and bits 6-4 are not significant
	6-4	an interrupt level (0-7)
	3-0	undefined

E\$OK, E\$NOT\$CONFIGURED

CALL RQ\$GET\$POOL\$ATTRIB (attrib\$ptr, except\$ptr);

PD

PW

Attrib:

STRUCTURE(POOL\$MAX	WORD,
POOL\$MIN	WORD,
INITIAL\$SIZE	WORD,
ALLOCATED	WORD,
AVAILABLE	WORD);

E\$OK, E\$NOT\$CONFIGURED

priority = RQ\$GET\$PRIORITY (task, except\$ptr);

B

WT

PW

E\$OK, E\$EXIST, E\$NOT\$CONFIGURED, E\$TYPE

size = RQ\$GET\$SIZE (segment, except\$ptr);

W

WT

PW

E\$OK, E\$EXIST, E\$NOT\$CONFIGURED, E\$TYPE

token = RQ\$GET\$TASK\$TOKENS (selection, except\$ptr);
WT B PW

selection	token
0	calling task
1	calling task's job
2	parameter object of calling task's job
3	root job

ESOK, ESPARAM

type\$code = RQ\$GET\$TYPE (object, except\$ptr);
W WT PW

ESOK, E\$EXIST, E\$NOT\$CONFIGURED

CALL RQ\$INSPECT\$COMPOSITE (extension, composite,
WT WT

token\$list\$ptr, except\$ptr);
PD PW

token\$list:
STRUCTURE(num\$slots WORD,
num\$used WORD,
tokens(*) WORD);

ESOK, E\$CONTEXT, E\$EXIST, E\$NOT\$CONFIGURED, E\$TYPE

object = RQ\$LOOKUP\$OBJECT (job, name, time\$limit,
WT WT PS W

except\$ptr);
PW

ESOK, E\$CONTEXT, E\$EXIST, E\$LIMIT, E\$NOT\$CONFIGURED,
ESPARAM, E\$TIME, E\$TYPE

token\$list = RQ\$OFFSPRING (job, except\$ptr);
WT WT PW

ESOK, E\$EXIST, E\$LIMIT, E\$MEM, E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$RECEIVE\$CONTROL (region, except\$ptr);
WT PW

E\$OK, E\$EXIST, E\$NOT\$CONFIGURED, E\$TYPE

object = RQ\$RECEIVE\$MESSAGE (mailbox, time\$limit,
WT WT W

response\$ptr, except\$ptr);
PWT PW

E\$OK, E\$EXIST, E\$NOT\$CONFIGURED, E\$TIME, E\$TYPE

value = RQ\$RECEIVE\$UNITS (semaphore, units, time\$limit,
W WT W W

except\$ptr);
PW

E\$OK, E\$EXIST, E\$LIMIT, E\$NOT\$CONFIGURED, E\$TIME, E\$TYPE

CALL RQ\$RESET\$INTERRUPT (level, except\$ptr);
W PW

level:	Bits	Value
	15-7	0
	6-4	the interrupt level (0-7)
	3	1
	2-0	0

Note that actual level n equals encoded level n8H.

E\$OK, E\$CONTEXT, E\$NOT\$CONFIGURED, E\$PARAM

CALL RQ\$RESUMESTASK (task, except\$ptr);
WT PW

E\$OK, E\$CONTEXT, E\$EXIST, E\$STATE, E\$TYPE

CALL RQ\$SEND\$CONTROL (except\$ptr);
PW

E\$OK, E\$CONTEXT, E\$NOT\$CONFIGURED

CALL RQ\$SEND\$MESSAGE (mailbox, object, response,
WT WT WT

except\$ptr);
PW

E\$OK, E\$EXIST, E\$MEM, E\$NOT\$CONFIGURED, E\$TYPE

ESOK, EEXIST, ESLIMIT, E\$NOT\$CONFIGURED, E\$TYPE

```
except $ptr);
PW
```

Exception\$Info:

```
STRUCTURE(EXCEPTION$HANDLER$OFFSET WORD,  
          EXCEPTION$HANDLER$BASE   WORD,  
          EXCEPTION$MODE             BYTE);
```

Exception\$Mode	Control to Exception Handler
0	Never
1	On programmer error only
2	On environmental condition only
3	On all exceptional conditions

ESOK, ESNOT\$CONFIGURED, ESPARAM

```
CALL RQ$SET$INTERRUPT (level, interrupt$task$flag,
```

interrupt\$handler, interrupt\$handler\$ds, except\$ptr),
PI WT PW

level:	Bits	Value
	15-7	0
	6-4	the interrupt level (0-7)
	3	1
	2-0	0

Note that actual level n equals encoded level $n8H$.

E\$OK, E\$CONTEXT, E\$NOT\$CONFIGURED, E\$PARAM

```
CALL RQ$SET$OS$EXTENSION (os$extension, start$address,
```

except\$ptr),
PW

ESOK, E\$CONTEXT, E\$NOT\$CONFIGURED, ESPARAM

CALL RQ\$SET\$POOL\$MINIMUM (new\$min, except\$ptr);
W PW

E\$OK, E\$LIMIT, E\$NOT\$CONFIGURED

CALL RQ\$SET\$PRIORITY (task, priority, except\$ptr);*
WT B PW

E\$OK, E\$CONTEXT, E\$EXIST, E\$LIMIT, E\$NOT\$CONFIGURED,
E\$TYPE

CALL RQ\$SIGNAL\$EXCEPTION (exception\$code, param\$num,
W B

stack\$ptr, reserved, reserved, except\$ptr);*
W W W PW

E\$OK, E\$NOT\$CONFIGURED

CALL RQ\$SIGNAL\$INTERRUPT (level, except\$ptr);
W PW

level:	Bits	Value
	15-7	0
	6-4	the interrupt level (0-7)
	3	1
	2-0	0

Note that actual level n equals encoded level n8H.

E\$OK, E\$CONTEXT, E\$NOT\$CONFIGURED, E\$PARAM

CALL RQ\$SLEEP (time\$limit, except\$ptr);
W PW

E\$OK, E\$NOT\$CONFIGURED, E\$PARAM

CALL RQ\$SUSPEND\$TASK (task, except\$ptr);
WT PW

E\$OK, E\$CONTEXT, E\$EXIST, E\$LIMIT, E\$TYPE

CALL RQ\$UNCATALOG\$OBJECT (job, name, except\$ptr);
WT PS PW

E\$OK, E\$CONTEXT, E\$EXIST, E\$NOT\$CONFIGURED, E\$PARAM,
E\$TYPE

CALL RQ\$WAIT\$INTERRUPT (level, except\$ptr);
W PW

level:	Bits	Value
	15-7	0
	6-4	the interrupt level (0-7)
	3	1
	2-0	0

Note that actual level n equals encoded level n8H.

ESOK, ESCONTEXT, ESNOT\$CONFIGURED, E\$PARAM

I/O System Calls

CALL RQ\$A\$ATTACH\$FILE (user, prefix, subpath, resp\$mbox,
WT WT PS WT
except\$ptr);
PW

Applies to all types of files. Priority of calling task must be in the range 32 to 255.

ESOK, E\$BAD\$CALL, ESCONTEXT, E\$EXIST, E\$FNEXIST, E\$FTYPE,
ESIO, E\$LIMIT, E\$MEM, E\$NOPREFIX, ESNOT\$CONFIGURED,
ESNOUSER, E\$PARAM, E\$TYPE

CALL RQ\$A\$CHANGE\$ACCESS (user, prefix, subpath, id, access,
WT WT PS W B
resp\$mbox, except\$ptr);
WT PW

Access:	Bit	Data File Access	Directory File Access
	0	Delete	Delete
	1	Read	Display
	2	Append	Add Entry
	3	Update	Change Entry
	4-7	Reserved	Reserved

Applies only to named data and directory files. Calling task, specifying a non-null path must have a priority in the range 32 to 255 and must either be the owner of the file or have change-entry access to the file's parent directory.

E\$OK, E\$BAD\$CALL, E\$CONTEXT, E\$EXIST, E\$FACCESS,
E\$FLUSHING, E\$FNEXIST, E\$FTYPE, E\$IFDR, E\$IO, E\$LIMIT,
E\$MEM, E\$NOPREFIX, E\$NOT\$CONFIGURED, E\$NOUSER,
E\$TYPE

CALL RQ\$A\$CLOSE (connection, resp\$mbx, except\$ptr);
WT WT PW

Applies to all types of files.

E\$OK, E\$BAD\$CALL, E\$CONTEXT, E\$EXIST, E\$FLUSHING,
E\$LIMIT, E\$MEM, E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$A\$CREATEDIRECTORY (user, prefix, subpath, access,
WT WT PS B

resp\$mbx, except\$ptr);
WT PW

Access:	Bit	Kind of Access
	0	Delete
	1	Display
	2	Add Entry
	3	Change Entry
	4-7	Reserved

Applies to named directory files only. Calling task must have a priority in the range 32 to 255 and must have add-entry access to the parent of the new directory.

E\$OK, E\$BAD\$CALL, E\$EXIST, E\$FACCESS, E\$FEXIST,
E\$FNEXIST, E\$FTYPE, E\$IFDR, E\$IO, E\$LIMIT, E\$MEM
E\$NOPREFIX, E\$NOT\$CONFIGURED, E\$NOUSER, E\$PARAM
E\$SPACE, E\$TYPE

CALL RQ\$A\$CREATE\$FILE (user, prefix, subpath, access,
WT WT PS B

granularity, high\$size, low\$size, must\$create, resp\$mbx,
W W W B W

except\$ptr);
PW

Access	Bit	Kind of Access
	0	Delete
	1	Read
	2	Append
	3	Update
	4-7	Reserved

Granularity	Value	Meaning
	0	Same as volume granularity
	0FFFFH	The file must be contiguous
	other	Number of bytes allocation

Applies to all types of files except named directory files. Priority of calling task must be in the range 32 to 255 and, for named files, calling task must have add-entry access to the new file's parent directory.

ESOK, ESBADSCALL, ESDCONTEXT, ESEXIST, ESFACCESS, ESFEXIST, ESFNEXIST, ESFTYPE, ESIO, ESLIMIT, ESMEM, ESNOPREFIX, ESNOTSCONFIGURED, ESNOUSER, ESSPACE, ESTYPE

CALL RQSASDELETESCONNECTION (connection resp\$mbx,
WT WT

except\$ptr);
PW

Applies to all types of files

ESOK, ESBADSCALL, ESDCONTEXT, ESEXIST, ESIO, ESLIMIT, ESMEM, ESNOTSCONFIGURED, ESTYPE

CALL RQSASDELETESFILE (user, prefix, subpath resp\$mbx,
WT WT PS WT

except\$ptr);
PW

Applies to stream and named files only. Calling task using non-null path must have priority 32 or higher

ESOK, ESBADSCALL, ESDCONTEXT, ESEXIST, ESFACCESS, ESFNEXIST, ESFTYPE, ESIFDR, ESIO, ESLIMIT, ESMEM, ESNOPREFIX, ESNOTSCONFIGURED, ESNOUSER, ESPARAM

CALL RQASGET\$CONNECTION\$STATUS (connection,

W

resp\$mbx, except\$ptr);

W

PW

Result (to resp\$mbx):

STRUCTURE (STATUS	WORD.
FILESDRIVER	BYTE.
FLAGS	BYTE.
OPENS MODE	BYTE.
SHARE	BYTE.
LOW\$FILES PTR	WORD.
HIGH\$FILES PTR	WORD.
ACCESS	BYTE).

File\$driver

Kind of Files Supported

1	Physical files
2	Stream files
4	Named files

Open\$mode

Connection Mode

0	Connection is closed
1	Open for reading
2	Open for writing
3	Open for reading and writing

Share

Extent of Sharing

0	Private use only
1	Share with readers only
2	Share with writers only
3	Share with all users

Access	Bit	Data File Access	Directory File Access
	0	Delete	Delete
	1	Read	Display
	2	Append	Add Entry
	3	Update	Change Entry
	4-7	Reserved	Reserved

Applies to all types of files

ESOK E\$BAD\$CALL E\$EXIST E\$FLUSHING E\$LIMIT E\$MEM
E\$NOT\$CONFIGURED E\$TYPE

GetC

CALL RQ\$ASGET\$DIRECTORY\$ENTRY (connection, entry\$Num,
WT W

resp\$mbx, except\$ptr);
WT PW

Result (to resp\$mbx):

STRUCTURE (STATUS WORD,
NAME(14) BYTE);

Applies to named directory files only.

ESOK, E\$BAD\$CALL, E\$DIR\$SEND, E\$EMPTY\$ENTRY, E\$EXIST,
E\$F\$ACCESS, E\$FLUSHING, E\$F\$TYPE, E\$IFDR, E\$IO, E\$LIMIT,
E\$MEM, E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$ASGET\$EXTENSION\$DATA (connection, resp\$mbx,
WT WT

except\$ptr);
PW

Result (to resp\$mbx):

STRUCTURE (STATUS WORD,
COUNT BYTE,
INFO(') BYTE);

ESOK, E\$EXIST, E\$IFDR, E\$IO, E\$LIMIT, E\$MEM,
E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$ASGET\$FILES\$STATUS (connection, resp\$mbx,
WT WT

except\$ptr);
PW

Result (to resp\$mbx):

STRUCTURE (STATUS WORD,
NUM\$CONN WORD,
NUM\$READER WORD,
NUM\$WRITER WORD,
SHARE BYTE,
NAMED\$FILE BYTE,
DEV\$NAME(14) BYTE,
FILE\$DRIVERS WORD,
FUNCTS WORD,
DEV\$GRAN WORD,
LOW\$DEV\$SIZE WORD,
HIGH\$DEV\$SIZE WORD,
DEV\$CONN WORD);

Get 0

Share**Extent of Sharing**

0	Private use only
1	Share with readers only
2	Share with writers only
3	Share with all users

File\$drivers:**Bit****Driver No.****Driver**

0	1	Physical file
1	2	Stream file
2	3	Reserved
3	4	Named file

Functs:**Bit****Function**

0	FSREAD
1	FSWRITE
2	FSSEEK
3	FSSPECIAL
4	FSATTACH\$DEV
5	FSDETACH\$DEV
6	FSOPEN
7	FSCLOSE
8-15	Reserved

Extra result (follows other structure):

STRUCTURE (FDESCSNUM

FILETYPE	WORD.
FILESGRAN	BYTE.
OWNER	BYTE.
LOWSCREATESTIME	WORD.
HIGHSCREATESTIME	WORD.
LOWSACCESSSTIME	WORD.
HIGHSACCESSSTIME	WORD.
LOW\$MODSTIME	WORD.
HIGH\$MODSTIME	WORD.
LOW\$FILESSIZE	WORD.
HIGH\$FILESSIZE	WORD.
LOW\$FILESBLOCKS	WORD.
HIGH\$FILESBLOCKS	WORD.
VOL\$NAME(16)	BYTE.
VOL\$GRAN	WORD.

LOWSVOLSSIZE	WORD.
HIGHSVOLSSIZE	WORD.
IDSCOUNT	WORD.
FIRST\$ACCESS	BYTE.
FIRST\$ID	WORD
SECONDS\$ACCESS	BYTE.
SECONDS\$ID	WORD.
THIRDS\$ACCESS	BYTE
THIRDS\$ID	WORD);

Applies to all types of files.

ESOK, ESBADSCALL, E\$EXIST, ESFLUSHING, ESLIMIT, E\$MEM,
ESNOT\$CONFIGURED, ESTYPE

CALL RQSASGET\$PATH\$COMPONENT (connection, resp\$mbx,
WT WT

except\$ptr);
PW

Result (to resp\$mbx):
STRUCTURE (STATUS WORD,
NAME(14) BYTE);

Applies to all types of files.

ESOK, ESBADSCALL, E\$EXIST, ESFLUSHING, ESIO, ESLIMIT,
E\$MEM, ESNOT\$CONFIGURED, ESTYPE

CALL RQSASOPEN (connection, mode, share, resp\$mbx,
WT B B WT

except\$ptr);
PW

Mode	Connection Mode
1	Open for reading
2	Open for writing
3	Open for both reading and writing
Share	Extent of Sharing
0	Private use only
1	Share with readers only
2	Share with writers only
3	Share with all users

Get P

Directory files may only be opened for reading and for sharing with all users.

ESOK, ESBADSCALL, ESCONTEXT, ESEXIST, ESFLUSHING,
ESLIMIT, ESMEM, ESNOTSCONFIGURED, ESPARAM,
ESSHARE, ESTYPE

CALL RQAS\$PHYSICAL\$ATTACH\$DEVICE (dev\$Name, file\$driver,
PS B
resp\$mbx, except\$ptr);
WT PW

File\$driver:	Value	File Driver
	1	Physical
	2	Stream
	4	Named

ESOK, ESCONTEXT, ESDEVFD, ESEXIST, ESFNEXIST, ESILLVOL,
ESIO, ESLIMIT, ESMEM, ESPARAM, ESTYPE

CALL RQAS\$PHYSICAL\$DETACH\$DEVICE (connection, hard,
WT B
resp\$mbx, except\$ptr);
WT PW

ESOK, ESCONTEXT, ESEXIST, ESIO, ESLIMIT, ESMEM,
ESNOTSCONFIGURED, ESTYPE

CALL RQAS\$READ (connection, buff\$ptr, count, resp\$mbx,
WT P W WT
except\$ptr);
PW

Applies to all types of files.

ESOK, ESBADSCALL, ESCONTEXT, ESEXIST, ESFLUSHING, ESIO,
ESLIMIT, ESMEM, ESNOTSCONFIGURED, ESTYPE

CALL RQAS\$RENAME\$FILE (connection, user, prefix, subpath,
WT WT WT PS
resp\$mbx, except\$ptr);
WT PW

Applies only to named data and directory files. Calling task must have delete access to the file and add-entry access to the file's new parent directory.

Ren

ESOK, ESBAD\$CALL, ESCONTEXT, E\$EXIST, E\$FACCESS,
E\$FEXIST, E\$FLUSHING, E\$FNEXIST, E\$FTYPE, E\$IFDR,
ESIO, ESLIMIT, ESMEM, ESNOPREFIX, E\$NOT\$CONFIGURED,
ESNOUSER, ESPARAM, ESTYPE

CALL RQ\$ASSEEK (connection, mode, hi\$ptr\$move,
WT B W

low\$ptr\$move, resp\$mbx, except\$ptr);
W WT PW

Mode	Action on Pointer
1	Backward by ptr\$move
2	Equal to ptr\$move
3	Forward by ptr\$move
4	To end-of-file minus ptr\$move

Applies only to physical files and named data files.

ESOK, ESBAD\$CALL, ESCONTEXT, E\$EXIST, E\$FLUSHING,
E\$IFDR, ESIO, ESLIMIT, ESMEM, E\$NOT\$CONFIGURED,
ESPARAM, ESTYPE

CALL RQ\$ASSET\$EXTENSIONS\$DATA (connection, data\$ptr,
WT PD

resp\$mbx, except\$ptr);
WT PW

data:

STRUCTURE (COUNT BYTE,
INFO(*) BYTE);

Applies to asynchronous connections created via the named file driver.

ESOK, E\$EXIST, ESIO, ESLIMIT, ESMEM, E\$NOT\$CONFIGURED,
ESPARAM, ESTYPE

CALL RQ\$ASSPECIAL (connection, spec\$func, ioparm\$ptr,
WT W P

resp\$mbx, except\$ptr);
WT PW

Seek

Spec\$func:

File Driver	Spec\$func	
for Connection	Value	Function
physical	0	format 204 or 206 track
physical or named	2	notify
stream	0	query
stream	1	satisfy

io\$param (for formatting 204 floppy track):

STRUCTURE (TRACK\$NUMBER WORD,
INTERLEAVE WORD,
TRACK\$OFFSET WORD);

io\$param (for formatting 206 hard disk track):

STRUCTURE (TRACK\$NUMBER WORD,
INTERLEAVE WORD,
TRACK\$OFFSET WORD,
FILLSCHAR WORD);

io\$param (for notify):

STRUCTURE (MAILBOX WORD,
OBJECT WORD);

ESOK, E\$BAD\$CALL, E\$CONTEXT, E\$EXIST, E\$FLUSHING,
E\$IDDR, E\$IFDR, E\$IO, E\$LIMIT, E\$MEM, E\$NOT\$CONFIGUREI
E\$TYPE

CALL RQ\$ASTRUNCATE (connection, resp\$mbx, except\$ptr);
WT WT PW

Applies to named data files only.

ESOK, E\$BAD\$CALL, E\$CONTEXT, E\$EXIST, E\$FACCESS,
E\$FLUSHING, E\$IFDR, E\$IO, E\$LIMIT, E\$MEM,
E\$NOT\$CONFIGURED, E\$TYPE

CALL RQ\$ASWRITE (connection, buff\$ptr, count, resp\$mbx,
WT P W WT

except\$ptr)
PW

Applies to all types of files except named directory files. The calling task must have append or update access to the file.

ESOK, ESBADSCALL, ESCONTEXT, ESEXIST, ESFACCESS
ESFLUSHING, ESIO, ESLIMIT, ESMEM, ESNOTSCONFIGURED,
ESSPACE, ESTYPE

user = RQSCREATESUSER (ids\$ptr, except\$ptr); *
WT PS PW

ids:
STRUCTURE (LENGTH WORD,
COUNT WORD,
ID(*) WORD);

ESOK, ESLIMIT, ESMEM, ESNOTSCONFIGURED, ESPARAM

CALL RQ\$DELETESUSER (user, except\$ptr); *
WT PW

ESOK, ESEXIST, ESNOTSCONFIGURED, ESTYPE

connection = RQ\$GET\$DEFAULT\$PREFIX (job, except\$ptr);
WT WT PW

ESOK, ESBADSCALL, ESEXIST, ESNOPREFIX, ESNOTS-
CONFIGURED, ESTYPE

user\$Id = RQ\$GET\$DEFAULT\$USER (job, except\$ptr);
WT WT PW

ESOK, ESBADSCALL, ESEXIST, ESNOTSCONFIGURED, ESNOUSER,
ESTYPE

date\$Time = RQ\$GET\$TIME (except\$ptr);
DW PW

ESOK, ESBADSCALL, ESNOTSCONFIGURED

CALL RQ\$INSPECT\$USER (user, ids\$ptr, except\$ptr); *
WT PS PW

ids:
STRUCTURE (LENGTH WORD,
COUNT WORD,
ID(*) WORD);

ESOK, ESEXIST, ESNOTSCONFIGURED, ESPARAM, ESTYPE

USER

CALL RQ\$SET\$DEFAULT\$PREFIX (job, prefix, except\$ptr);
WT WT PW

ESOK, E\$BAD\$CALL, E\$EXIST, E\$LIMIT, E\$MEM, E\$NOTS-
CONFIGURED, E\$TYPE

```
CALL RQ$SET$DEFAULT$USER (job, user, except$ptr);
                        WT WT      PW
```

ESOK, ESBAD\$CALL, E\$EXIST, ESLIMIT, E\$MEM, E\$NOT\$-
CONFIGURED, E\$TYPE

CALL RQ\$SET\$TIME (time\$high, time\$low, except\$ptr); *

ESOK, E\$NOT\$CONFIGURED

LOADER CALL

```
CALL RQSASLOAD (connection, resp$mbbox, except$ptr);
```

Result (to resp\$mbbox):

STRUCTURE (STATUS	WORD,
RECORD\$COUNT	WORD
ERROR\$RECSTYPE	BYTE,
NUM\$UNDEF\$REFS	WORD,
INIT\$IP	WORD,
INIT\$CS	WORD,
STACK\$OFFSET	WORD,
INIT\$SS	WORD,
STACK\$SIZE	WORD,
INIT\$DS	WORD)

ESOK, ESABSSADDRESS, ESBADSGRP, ESBAD\$HDR, ESBAD\$SEG,
ESCHECKSUM, ESCONTEXT, E\$EXIST, E\$NOT\$CONFIGURED,
ESRECSFMT, ESRECSLENGTH, ESREC\$TYPE, ESSEG\$ALLOC,
ESTYPE

I/O SYSTEM RESULT SEGMENT

STRUCTURE	(STATUS	WORD,
	UNIT\$STATUS	WORD,
	ACTUAL	WORD,
	ACTUAL\$FILL	WORD,
	DEVICE	WORD,
	UNIT	BYTE,
	FUNC	BYTE,
	SPEC\$FUNC	WORD,
	LOW\$DEVSLOC	WORD,
	HIGH\$DEVSLOC	WORD,
	BUFF\$PTR	POINTER,
	LOW\$COUNT	WORD,
	HIGH\$COUNT	WORD,
	AUX\$PTR	POINTER,
	LINK\$FOR	POINTER,
	LINK\$BACK	POINTER,
	RESP\$MBOX	WORD,
	DONE	BYTE);

Values in low-order byte of UNIT\$STATUS
(valid only when STATUS = ESIO = 2BH):

Value	Mnemonic	Meaning
0	IOSUNCLASS	Unclassified error
1	IOSOFT	Soft error; retry is possible
2	IOSHARD	Hard error; retry not possible
3	IOSOPRINT	Operator intervention required
4	IOSWRPROT	Write-protected volume

I/O SYSTEM PATH/SUBPATH INFORMATION

Prefix Parameter	Subpath Parameter	Designated Connection
0	Either 0 or pointer to null string	Connection whose token is the default prefix
0	Pointer to ASCII string	ASCII string defines a path from the connection whose token is the default prefix to the target connection
token	Either 0 or pointer to null string	Connection whose token is contained in the prefix parameter
token	Pointer to ASCII string	Prefix parameter contains a token for connection. ASCII string defines a path from that connection to the target connection

TERMINAL HANDLER

SPECIAL CHARACTERS AND THEIR EFFECTS

Special Character	Effect
RUBOUT	Deletes previously entered character.
Carriage Return	Signals end of line.
Line Feed	Signals end of line.
ESCAPE	Signals end of line.
control-C	Calls the RQ\$ABORT\$AP procedure.
control-D	Activates the Debugger.
control-O	Kills or restarts output.
control-Q	Resumes suspended non-Debugger output.
control-R	Displays current line with editing.
control-S	Suspends non-Debugger output.
control-X	Deletes the current line.
control-Z	Sends an empty message.

REQUEST MESSAGE FORMAT

Offset	Field
0	Function
2	Count
4	Exception Code
6	Actual
8	Message Content

INTERRUPT INFORMATION

ALLOCATION OF INTERRUPT VECTORS

- 0- 55: reserved
- 56- 63 available for external interrupt levels 0-7, respectively
- 64-223: reserved
- 224-255: available for Operating System Extensions
1-32, respectively

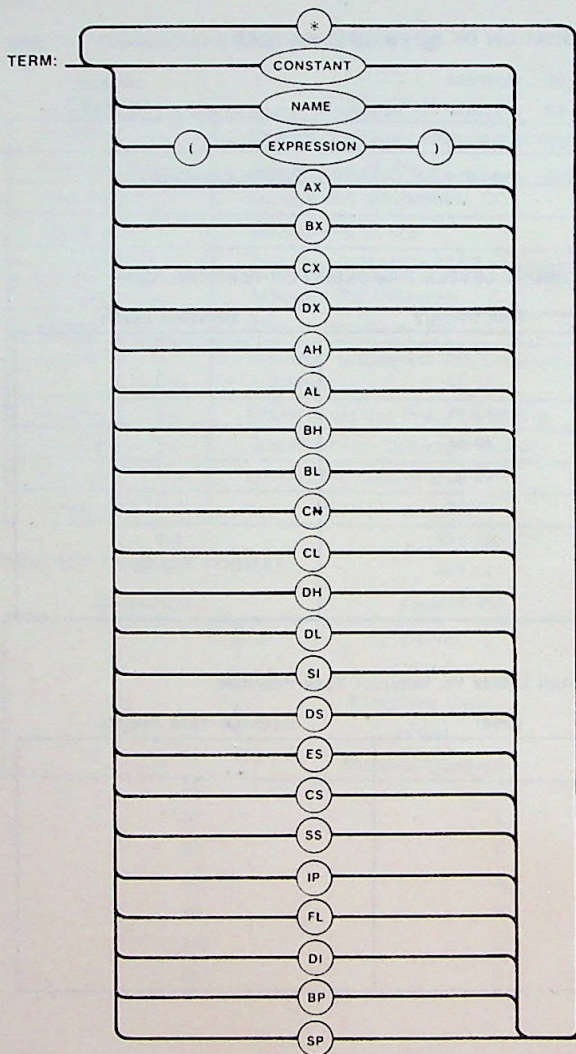
INTERRUPT LEVELS DISABLED FOR RUNNING TASK

Task Priority	Disabled Levels
0-16	0-7
17-32	1-7
33-48	2-7
49-64	3-7
65-80	4-7
81-96	5-7
97-112	6-7
113-128	7
129-255	None

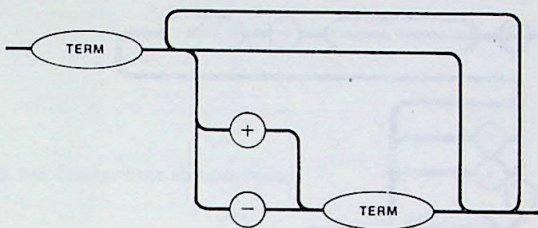
Interrupt Levels Vs. Interrupt Task Priorities

Level	Interrupt Task Priority
0	18
1	34
2	50
3	66
4	82
5	98
6	114
7	130

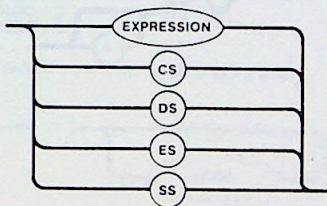
DEBUGGER SYNTAX



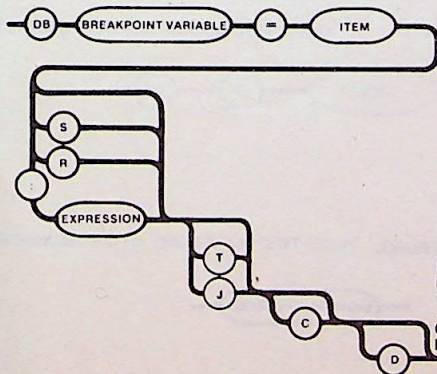
EXPRESSION:



ITEM:

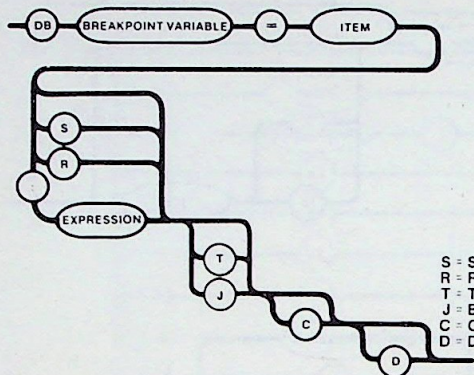


ESTABLISHING A BREAKPOINT:

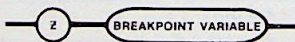


S = SENDERS ONLY
R = RECEIVERS ONLY
T = TASK ONLY
J = ENTIRE JOB
C = CONTINUE
D = DELETE

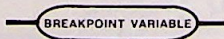
CHANGING A BREAKPOINT:



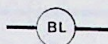
DELETING A BREAKPOINT:



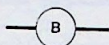
EXAMINING A BREAKPOINT:



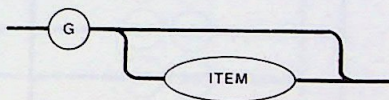
VIEWING THE BREAKPOINT LIST:



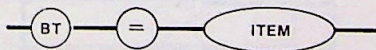
VIEWING THE BREAKPOINT PARAMETERS:



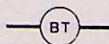
REMOVING A TASK FROM THE BREAKPOINT LIST:



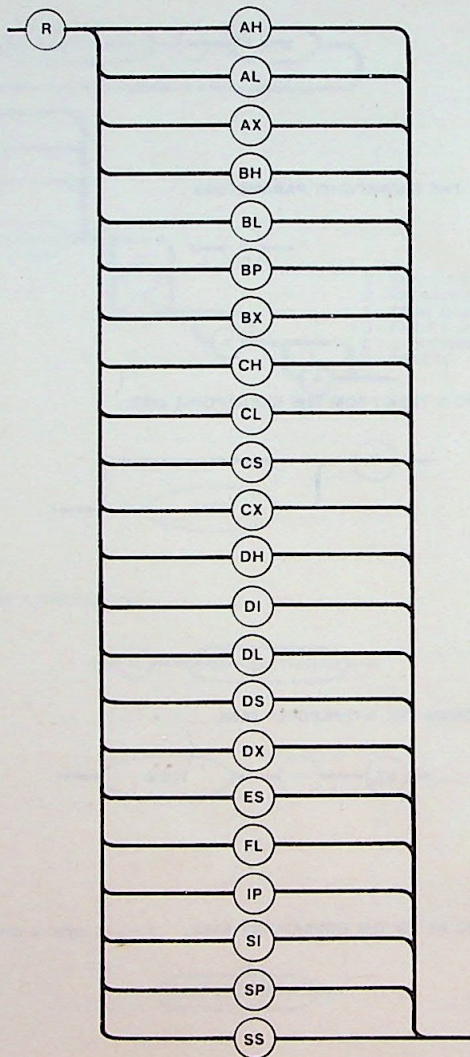
ESTABLISHING THE BREAKPOINT TASK:



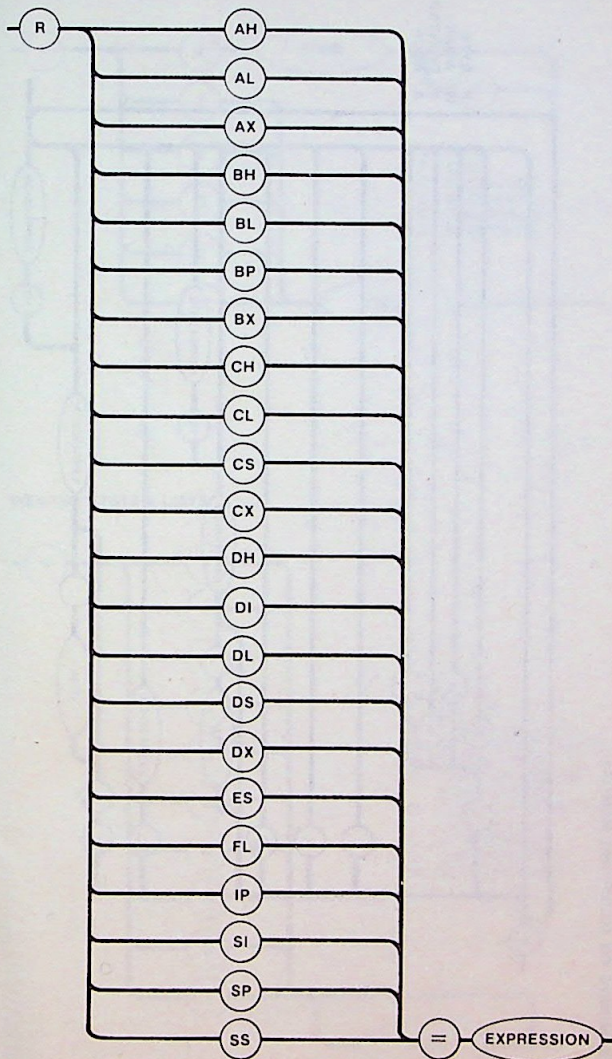
INQUIRING AS TO THE BREAKPOINT TASK:



VIEWING THE BREAKPOINT TASK'S REGISTERS:

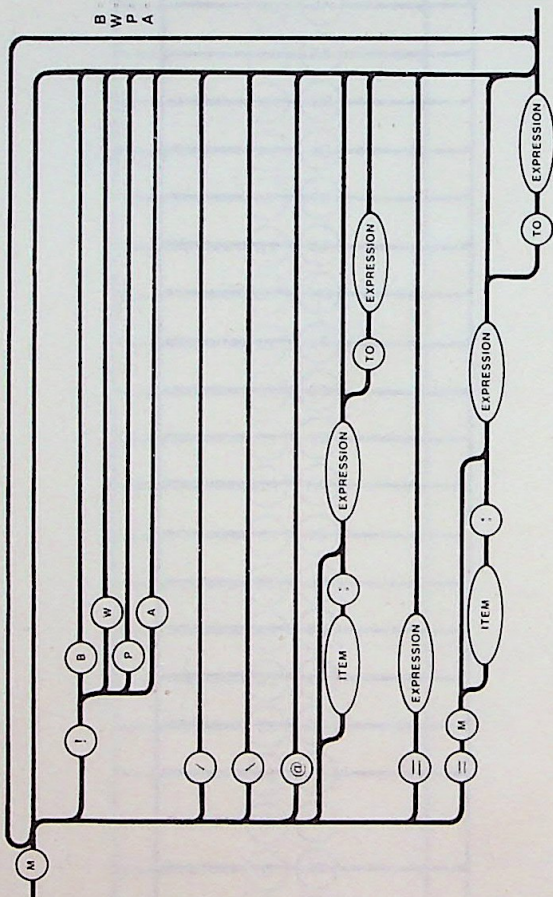


ALTERING THE BREAKPOINT TASK'S REGISTERS:

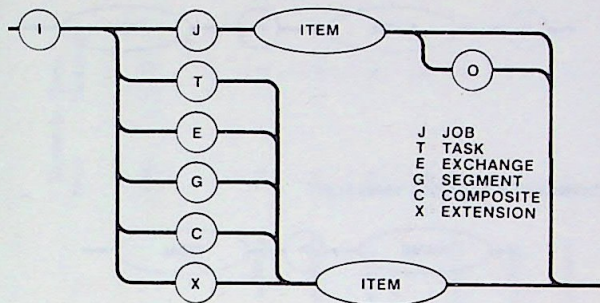


EXAMINING OR MODIFYING MEMORY:

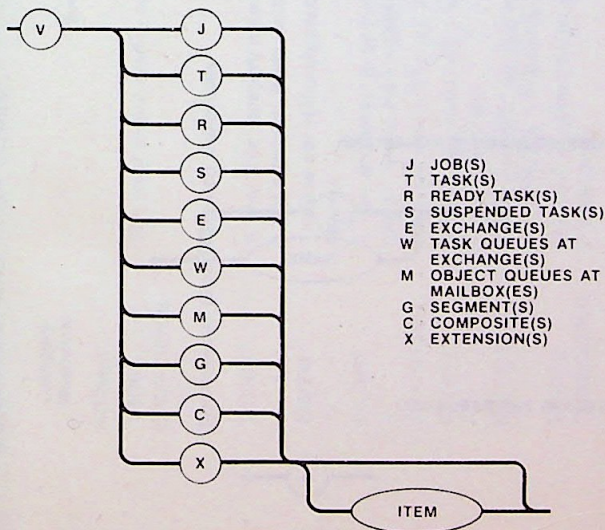
B - BYTE
W - WORD
P - POINTER
A - ASCII



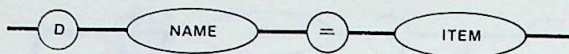
INSPECTING SYSTEM OBJECTS:



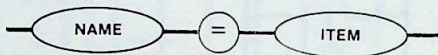
VIEWING SYSTEM LISTS:



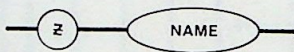
DEFINING A NUMERIC VARIABLE:



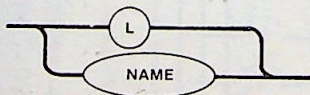
CHANGING A NUMERIC VARIABLE:



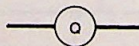
DELETING A NUMERIC VARIABLE:



VIEWING NUMERIC VARIABLE(S):



EXITING THE DEBUGGER:



CONDITIONS AND THEIR CODES

Category/ Mnemonic	Meaning	Numeric Code	
		Hex	Decimal
NORMAL			
ESOK	The most recent system call was successful	0H	0
EXCEPTIONAL			
Synchronous (Environmental) Conditions			
ESTIME	A time limit (possibly a limit of zero time) expired without a task's request being satisfied.	1H	1
ESMEM	There is not sufficient memory available to satisfy a task's request.	2H	2
ESLIMIT	A task attempted an operation which, if it had been successful, would have violated a Nucleus-enforced limit.	4H	4
ESCONTEXT	A system call was issued out of proper context.	5H	5
ESEXIST	A token parameter has a value which is not the token of an existing object.	6H	6
ESSTATE	A task attempted an operation which would have caused an impossible transition of a task's state	7H	7
ESNOTS-CONFIGURED	The most recently issued system call is not in the present configuration.	8H	8

Category/ Mnemonic	Meaning	Numeric Code	
		Hex	Decimal
ESFEXIST	The prefix and subpath arguments specify a file that already exists	20H	32
ESFNEXIST	The prefix and subpath arguments do not specify an existing file	21H	33
ESSUPPORT	The given combination of parameters is not supported	23H	35
ESFACCESS	Access to the file is denied	26H	38
ESFTYPE	The specified file is not of the correct type for this system call	27H	39
ESSPACE	The available space on the specified volume is not sufficient to satisfy the request	29H	41

Synchronous (Programmer Error) Conditions

ESZEROS-DIVIDE	A task attempted to divide by zero.	8000H	32768
ESOVERFLOW	An overflow interrupt occurred.	8001H	32769
ESTYPE	A token parameter referred to an existing object that is not of the required type.	8002H	32770
ESPARAM	A parameter which is neither a token nor an offset has an illegal value.	8004H	32772
ESBADSCALL	The I/O System code has been damaged fatally	8005H	32773

ESIFDR	The request is not valid for files supported by the file driver implied in the request	8020H	32800
ESNOUSER	The calling task's job does not have a default user object	8021H	32801
ESNOPREFIX	The calling task's job does not have a default prefix	8022H	32802

Asynchronous (I/O) Conditions

ESMEM	There is not sufficient memory available to satisfy a task's request	2H	2
ESLIMIT	A task attempted an operation which, if it had been successful, would have violated a Nucleus-enforced limit.	4H	4
ESCONTEXT	A system call was issued out of proper context.	5H	5
ESFEXIST	The prefix and subpath arguments specify a file that already exists	20H	32
ESFNEXIST	The prefix and subpath arguments do not specify an existing file	21H	33
ESDEVFD	The specified device is not compatible with the specified file driver	22H	34
ESSUPPORT	The given combination of parameters is not supported	23H	35
EEMPTYSENTRY	The specified file has been deleted and the I/O System has not reissued the entry to another file	24H	36
ESDIRSEND	A parameter points beyond the end of a directory	25H	37

Category/ Mnemonic	Meaning	Numeric Code	
		Hex	Decimal
ESFACCESS	Access to the file is denied.	26H	38
ESFTYPE	The specified file is not of the correct type for this system call	27H	39
ESSHARE	The request is not compatible with the current sharing status of the specified file	28H	40
ESSPACE	The available space on the specified volume is not sufficient to satisfy the request	29H	41
ESIDDR	The request is not valid for the device driver implied in the request	2AH	42
ESIO	An I/O error occurred during the operation	2BH	43
ESFLUSHING	The connection implied in the call was deleted before the operation was completed	2CH	44
Asynchronous (Loader) Conditions			
ESBAD\$HDR	The header record in the specified file was invalid	62H	98
ESCHECKSUM	A checksum error occurred during loading	64H	100
E\$EOF	The Loader encountered an end-of-file	65H	101
E\$FIXUP	The Loader encountered either an invalid fixup record or a fixup mode that cannot be handled	66H	102
ESNO\$MEM	There is not sufficient memory available to run the Loader	67H	103

ESRECSFMT	There was an unspecified error in an object record	69H	105
ESRECS- LENGTH	An object record was too long for the Loader's internal buffer	70H	106
ESRECSTYPE	The specified object record type was invalid	71H	107
ESNOSTART	A starting address is not specified in the object module	72H	108

SUBJECT INDEX

Condition Codes	37
Debugger Syntax	28
Interrupt Levels Disabled for Running Task	27
Interrupt Levels vs. Interrupt Task Priorities	27
Interrupt Vector Allocation	27
I/O System Calls	12
I/O System Path/Subpath Information	25
I/O System Result Segment	24
Loader Call	23
Nucleus Calls	2
Terminal Handler Request Message Format	26
Terminal Handler Special Characters	26